

Abstract for Flight Software Workshop – 2016

Title: Automation of Flight Software Regression Testing

List of Authors: Scott B. Tashakkor, P. E., Marshall Space Flight Center, NASA

NASA is developing the Space Launch System (SLS) to be a heavy lift launch vehicle supporting human and scientific exploration beyond earth orbit. SLS will have a common core stage, an upper stage, and different permutations of boosters and fairings to perform various crewed or cargo missions. Marshall Space Flight Center (MSFC) is writing the Flight Software (FSW) that will operate the SLS launch vehicle. The FSW is developed in an incremental manner based on “Agile” software techniques. As the FSW is incrementally developed, testing the functionality of the code needs to be performed continually to ensure that the integrity of the software is maintained. Manually testing the functionality on an ever-growing set of requirements and features is not an efficient solution and therefore needs to be done automatically to ensure testing is comprehensive. To support test automation, a framework for a regression test harness has been developed and used on SLS FSW. The test harness provides a modular design approach that can compile or read in the required information specified by the developer of the test. The modularity provides independence between groups of tests and the ability to add and remove tests without disturbing others. This provides the SLS FSW team a time saving feature that is essential to meeting SLS Program technical and programmatic requirements.

During development of SLS FSW, this technique has proved to be a useful tool to ensure all requirements have been tested, and that desired functionality is maintained, as changes occur. It also provides a mechanism for developers to check functionality of the code that they have developed. With this system, automation of regression testing is accomplished through a scheduling tool and/or commit hooks. Key advantages of this test harness capability includes execution support for multiple independent test cases, the ability for developers to specify precisely what they are testing and how, the ability to add automation, and the ability of the harness and cases to be executed continually. This test concept is an approach that can be adapted to support other projects.